

A Few Good Launch Coaches

Welcome to “Backfire,” the interview within a journal where we cross-examine popular icons for software truth. This month we have Col. Nathan R. Jessep, former commanding officer, Marine Ground Forces, Guantanamo Bay, Cuba, from the movie “A Few Good Men.” After a dishonorable discharge, Mr. Jessep has started a new career as a Team Software ProcessSM (TSPSM) Launch Coach.

Petersen: Welcome Mr. Jessep.

Jessep: Colonel.

Petersen: What’s that?

Jessep: I would appreciate it if you would address me as Colonel or Sir. I believe I have earned it.

Petersen: OK. Colonel, let us focus on your new career. Why did you implement Personal Software ProcessSM (PSPSM) and TSP on your project?

Jessep: I felt that our agile developers might be in danger without disciplined measurement.

Petersen: Grave danger?

Jessep: Is there another kind?

Petersen: Is it true you implemented TSP to drive out agile developers?

Jessep: No, I specifically gave orders for project managers to coddle developers with agile tendencies.

Petersen: Any chance they ignored that order?

Jessep: Ignored the order?

Petersen: Any chance they forgot about it?

Jessep: No.

Petersen: Any chance they left your office and thought, “The old man is wrong?”

Jessep: Have you ever spent time in a TSP unit, son?

Petersen: No, sir.

Jessep: Ever calculated earned value?

Petersen: No, sir.

Jessep: Ever put your project in another man’s hands; ask him to put his proj-

ect in yours?

Petersen: No, sir.

Jessep: We follow processes, son. We follow processes or projects die. It is that simple. Are we clear?

Petersen: Yes, sir.

Jessep: Are we clear!

Petersen: Crystal.

Petersen: Colonel, if you have sound processes and developers always follow those processes, why did you order managers to coddle agile programmers? Why the extra order?

Jessep: The agile developers were sub-standard programmers –

Petersen: But that is not what you said. You said your agile developers might be in danger without disciplined measurement. I said, ‘Grave danger.’ You said –

Jessep: I know what I said.

Petersen: Then why the order sir?

Jessep: Sometimes men take matters into their own hands.

Petersen: No, sir. You made it clear that your men never take matters into their own hands. Your men follow processes or projects die. Therefore, the project should not have been in any danger, Colonel.

Jessep: You pretentious wimp. You want answers?

Petersen: I think I am entitled to them.

Jessep: You want answers!

Petersen: I want the truth.

Jessep: You can’t handle the truth!

Jessep: Son, we live in a world run by software and that software has to be developed by men with discipline. Who is going to do it? You, Petersen? Bill Gates? I have a greater responsibility than you can possibly fathom. You weep for agile developers and you curse PSP. You have that luxury. You have the luxury of not knowing what I know: That the agile developers’ dismissal, while tragic, probably saved that project and my existence, while grotesque and incomprehensible to you, saves projects.

You don’t want the truth. Because deep down, in places you don’t talk about at parties, you want me developing software. You need my software. We use words like process, measurement, maturity ... we use these words as the backbone of a life spent developing software. You use them as a punch line.

I have neither the time nor the inclination to explain myself to a man who rises and sleeps under the blanket of the very software I provide, and then questions the manner in which I provide it. I would prefer you just said thank you and went on your way. Otherwise, I suggest you pick up a keyboard and code a module. Either way, I don’t give a darn what you think you are entitled to.

Petersen: Did you order their dismissal?

Jessep: I did my job and I would do it again.

Petersen: Did you order their dismissal!

Jessep: You’re darn right I did!

OK, thank you, Colonel Jessep. Join us next time when “Backfire” cross-examines the cast of Seinfeld on capability maturity models.

— Gary A. Petersen
Shim Enterprise, Inc.

Can You BACKTALK?

Here is your chance to make your point, even if it is a bit tongue-in-cheek, without your boss censoring your writing. In addition to accepting articles that relate to software engineering for publication in CROSSTALK, we also accept articles for the BACKTALK column. BACKTALK articles should provide a concise, clever, humorous, and insightful perspective on the software engineering profession or industry or a portion of it. Your BACKTALK article should be entertaining and clever or original in concept, design, or delivery. The length should not exceed 750 words.

For a complete author’s packet detailing how to submit your BACKTALK article, visit our Web site at <www.stsc.hill.af.mil>.